

WEST

☐ Generate Collection

L4: Entry 11 of 20

File: JPAB

Jan 16, 1998

PUB-NO: JP410011282A

DOCUMENT-IDENTIFIER: JP 10011282 A

TITLE: INSTALLING SYSTEM AND EXECUTING SYSTEM FOR SOFTWARE

PUBN-DATE: January 16, 1998

INVENTOR-INFORMATION:

NAME

COUNTRY

ARAI, YOSHIO

ASSIGNEE-INFORMATION:

NAME

COUNTRY

NEC ENG LTD

APPL-NO: JP08160315

APPL-DATE: June 20, 1996

INT-CL (IPC): G06 F 9/06; G06 F 9/06; G06 F 12/14

ABSTRACT:

PROBLEM TO BE SOLVED: To provide a software installing system preventing the illegal use of a software stored in an installing medium.

SOLUTION: The software installing medium 1 is provided with ID storing areas 12 to 14 storing ID showing the presence/absence of the execution of installing, ID showing the kind of the software, etc., and the nonvolatile memory and the secondary storage device 28 of a computer system 2 are provided with areas 23 to 25 and 281 to 283 for storing the same ID as these. At the time of installing the software, each ID stored in the medium 1 and IDs stored in the storage areas 23 to 25 of the nonvolatile memory are compared with each other to decide whether installing is to be executed corresponding to this comparing result or not.

COPYRIGHT: (C)1998,JPO

* NOTICES *

Japan Patent Office is not responsible for any damages caused by the use of this translation.

1. This document has been translated by computer. So the translation may not reflect the original precisely.
2. **** shows the word which can not be translated.
3. In the drawings, any words are not translated.

DETAILED DESCRIPTION

[Detailed Description of the Invention]

[0001]

[The technical field to which invention belongs] this invention relates to the prevention technology of the illegal copy protection for preventing that especially software copies unjustly and is used about the method which installs in a electronic computing system software stored in the storage (it is the same storing with the gestalt which can be performed, and the following), and double installation.

[0002]

[Description of the Prior Art] A user is provided with the program used installing in a computer system, application software, etc. by FD (flexible disk) and CD-ROM (compact disk ROM) which are an installation medium (software is called hereafter), and they are developed by the secondary storage of a user's computer system etc. by the installation means. However, it may be installed in another computer system by the malicious user if the installation medium is left as it is. For this reason, destroyed the installation medium conventionally, the illegal copy was prevented by managing by the software provider side, or the identification number storage section was prepared in the computer system, the identification number memorized to the identification number storing field of ID within a computer system and an installation medium was compared, and the method of preventing an unjust copy by these coincidence inequalities was taken (refer to Provisional-Publication-No. No. 213027 [63 to] official report).

[0003] However, when an installation medium is destroyed or the installed software carries out destruction etc. according to a certain cause by the method by which a software provider manages an installation medium, there is a problem which a regular user cannot cope with promptly. Moreover, the method indicated by the Provisional-Publication-No. No. 213027 [63 to] official report is a method which stores the ID also in a computer system side while it prepares the field which memorizes an identification number to an installation-medium side and stores an identification number in this identification number storage region in the case of installation. By this method, when ID is stored in the installation-medium side, it is judged as a used installation medium, and only when ID stored in the computer system is in agreement, re-installation is permitted. However, there were the following problems by this method.

[0004] (1) When an installation medium only with the number intact to a computer system [finishing / accidentally / installation / already when an installation medium is offered] is regularly used to two or more computer systems, re-installation is performed and the reuse of the misused installation medium cannot be carried out.

(2) When two or more software is memorized by large capacity storage, for example, CD-ROM etc., and a user is provided with it, a user cannot install the software to need alternatively.

(3) Correspondence when software upgrades cannot be performed.

[0005] Then, the technical problem of this invention is to offer the improved software installation method which can prevent the situation which can prevent software from an unjust copy, securing backup, and installs the same software in a computer system doubly accidentally. Other technical

problems of this invention are to offer the real whereabouts formula of the software installed by the above-mentioned method.

[0006]

[Means for Solving the Problem] The installation method of the software of this invention It is the method which installs the software stored in the storage in the storage of the computer system which has non-volatile memory. The installation ID storing field which stores the installation ID which expresses the existence of installation execution of the aforementioned software with the aforementioned storage, While forming the software ID storing field which stores the software ID which shows the classification of software, and the version ID storing field which stores the version ID which shows the version of the software concerned To the aforementioned non-volatile memory, the aforementioned installation ID, the aforementioned software ID And an installation ID generating means to form the storage region of the aforementioned version ID and to generate the aforementioned installation ID in the aforementioned computer system further at the time of installation of the beginning of the aforementioned software, While writing the generated installation ID in the installation ID storing field of the aforementioned storage, and the installation ID storage region of the aforementioned non-volatile memory Read the software ID which corresponds from the aforementioned software ID storing field, and it writes in the software ID storage region of the aforementioned non-volatile memory. Furthermore, it is characterized by establishing the means which reads the version ID which corresponds from the aforementioned version ID storing field, and is written in the version ID storage region of the aforementioned non-volatile memory.

[0007] By the installation method of the software of this invention, the storage region of the aforementioned installation ID, the aforementioned software ID, and the aforementioned version ID is further formed in the storage of the aforementioned computer system. And in case the aforementioned software is installed in a computer system, consistency with each ID memorized by each ID stored in the aforementioned storage and each aforementioned storage region is verified, and a means to determine the execution propriety of the aforementioned installation based on this verification result is established.

[0008] In addition, as for the aforementioned computer system, it is desirable to have an interface means to display a predetermined message in the case of installation of the aforementioned software, and to receive the response result of this message, and to determine the execution propriety of installation based on this received response result.

[0009] Other installation methods of this invention are methods which install the software stored in the storage in the storage of the computer system which has non-volatile memory, and are equipped with the common medium in which accessible rewriting is free to the aforementioned storage. And the aforementioned installation ID storing field is formed in this common medium at least. While forming the software ID storing field which stores the aforementioned software ID in the aforementioned storage, and the version ID storing field which stores Version ID To the aforementioned non-volatile memory, the aforementioned installation ID, the aforementioned software ID And an installation ID generating means to form the storage region of the aforementioned version ID and to generate the aforementioned installation ID in the aforementioned computer system further at the time of installation of the beginning of the aforementioned software, While writing the generated installation ID in the installation ID storing field of the aforementioned common medium, and the installation ID storage region of the aforementioned non-volatile memory Read the software ID which corresponds from the aforementioned software ID storing field, and it writes in the software ID storage region of the aforementioned non-volatile memory. Furthermore, it is characterized by establishing the means which reads the version ID which corresponds from the aforementioned version ID storing field, and is written in the version ID storage region of the aforementioned non-volatile memory.

[0010] Moreover, the real whereabouts formula of the software of this invention is a real whereabouts formula of the software installed in the storage of the computer system which has non-volatile memory. While forming in the aforementioned storage the aforementioned installation ID storage region, a software ID storage region, and the version ID storage region that memorizes Version ID The aforementioned installation ID inputted into the aforementioned non-volatile memory at the time of

installation of the aforementioned software The storage region of the aforementioned software ID and the aforementioned version ID is formed. Furthermore, it is characterized by establishing a means to verify the consistency of each ID stored in each aforementioned storage region, and each ID stored in the aforementioned non-volatile memory to the aforementioned computer system, and to determine the execution propriety of the aforementioned software as it based on this verification result.

[0011]

[Embodiments of the Invention] The form of operation of this invention is explained in detail with reference to a drawing. Drawing 1 is the block diagram of 1 operation form of this invention. This operation form shows an example in case the software registered into a computer system and the install program which performs installation are stored in the same storage (FD). In addition, the above-mentioned storage is hereafter called an installation software medium.

[0012] A sign 1 is the software installation medium 1 among drawing, and in case the storing field of software 11, the software ID storing field 12 where ID which shows the classification of this software 11 is stored, the version ID storing field 13 where the version ID of this software is stored, and this software 11 are installed in a computer system 2, the installation ID storing field 14 where the installation ID which shows that installation was performed is stored is formed.

[0013] The computer system 2 is equipped with a well-known installation means 3 to install software, the secondary storage 28 which stores the software installed, and the loader 26 which performs loading of software. The identification number comparison means 27 is formed in the loader 26. The storing field of software 29, the software ID storage region 281, the version ID storage region 282, and the installation ID storage region 283 are formed in the secondary storage 28. An identification number I/O means 22 by which a computer system 2 performs input/output control of the installation identification number generated with an installation identification number generation means 21 to generate an installation identification number again, and this installation identification number generation means 21, and the non-volatile memory (following, BUM) which is not illustrated are prepared, and the installation ID storage section 23, the software ID storage section 24, and the version ID storage section 25 are formed in this BUM.

[0014] Neither of ID is memorized by the installation ID storage section 23 in BUM, the software ID storage section 24, and the version ID storage section 25 in the initial state in which software is not installed. Moreover, at the time of the beginning by which the user etc. was provided with the software installation medium 1, no ID is written in the installation ID storing field 14 within software 11, but when software is installed first, ID is written in for the first time. On the other hand, ID of the software ID storing field 12 and ID of the version ID storing field 13 are the information on software 11 the very thing, and ID is written in when provided for a user.

[0015] It is dependent on combination -- whether ID is recorded on the installation ID storage section 23 in BUM, the software ID storage section 24, the version ID storage section 25, and the installation ID storing field 14 on the software installation medium 1 -- whether the software stored in the software installation medium 1 is installed in a computer system 2.

[0016] The processing corresponding to each combination is as follows.

(1) When no ID of the installation ID storage section 23 in BUM, the software ID storage section 24, the version ID storage section 25, and the installation ID storing field 14 on the software installation medium 1 has been stored. In this case, in the initial state in which software is not installed, it is the case where a computer system 2 is regularly provided with the software installation medium 1. If set in the I/O device which the software installation medium 1 does not illustrate, the identification number I/O means 22 will start. Drawing 2 is explanatory drawing of the procedure after starting of the identification number I/O means 22. If drawing 2 is referred to, the identification number I/O means 22 will acquire Software ID from the software ID storing field 12 first (S1). Next, it verifies whether it is in agreement with the software ID which read the software ID of the software ID storage section 24 within a computer system 2 (S2), and was read by S1 (S3). In this case, since Software ID is not in agreement, new registration processing is performed (S4). The procedure of this new registration processing is as being shown in drawing 3 , and the identification number I/O means 22 processes S11-S18 which are

shown in drawing 3 . In this example, since Installation ID has not been stored in the installation ID storage section 23 and the installation ID storing field 14, S13 is performed.

[0017] In S13, Installation ID is newly generated by the installation identification number generation means 21. Installation ID generates the time which was going to install, and a random number, and is generated. Namely, what is necessary is just to become the number of a meaning by many computer systems. The installation identification number generation means 21 will pass this to the identification number I/O means 22, if an installation identification number is generated. The identification number I/O means 22 answers this, and stores this installation identification number in the installation ID storing field 14 and the installation ID storage section 23, respectively (S14).

[0018] Next, the identification number I/O means 22 acquires Version ID from the version ID storing field 13, and stores it in the version ID storage section 25 while it stores in the software ID storage section 24 the software ID acquired by S1 (S17). ** [the identification number I/O means' 22 completion of storing processing of the above-mentioned identification number / start / the installation means 3 / means] The installation means 3 installs software 11 in a secondary storage 28 (S18). The software 29 which memorized the same installation ID as the installation ID storage section 23 is stored in a secondary storage 28 by the above. Then, if use of software 29 is started by the computer system 2 and a loader 26 is started, loader processing shown in drawing 6 will be performed.

[0019] If drawing 6 is referred to, while a loader 26 reads each ID from the installation ID storage region 283 of the software 29 by which the load demand was carried out, the software ID storage region 281, and the version ID storage region 282 first ID which is in agreement with the software ID of the software ID storage section 24 to the software 29 in Installation ID from the installation ID storage section 23 The version ID which is the software ID read from the previous software ID storage section 24 and a pair from the version ID storage section 25 is read (S41), and it distinguishes whether each ID of both is in agreement by the identification number comparison means 27 (S42). Any of Installation ID, Software ID, and Version ID, or when at least one is not in agreement, load processing is not performed, but execution of software is made to terminate abnormally (S43). In this example, since all ID is in agreement, software 29 is loaded and performed (S44).

[0020] (2) In the case of the combination with which have not ID stored in the installation ID storing field 14, and the same ID as the software ID of the software ID storing field 12 is not remembered to be by the software ID storage section 24 although each ID is stored in the installation ID storage section 23, the software ID storage section 24, and the version ID storage section 25 That is, it is the case where another software uses the software installation medium 1 offered regularly for the already installed computer system 2, and installs software 11 in it newly. In this case, like the case where an example (1) shows, although Software ID is acquired by S1 and S2 in the procedure view shown in drawing 2 and the consistency of ID is verified by S3, since there is no same ID as the software ID of the software ID storing field 12 into the software ID storage section 24, both are not in agreement and new registration processing shown in drawing 3 is performed in this example (2).

[0021] In new registration processing of drawing 3 , since Installation ID exists in the installation ID storage section 23 and there is no installation ID in the installation ID storing field 14, in S16, the installation ID of the installation ID storage section 23 is registered into the installation ID storing field 14. Here, the identification number I/O means 22 does not newly generate Installation ID with the installation identification number generation means 21, but reads the installation ID already memorized by the installation ID storage section 23, and stores it in the installation ID storing field 14. Next, Software ID and Version ID are acquired from the software installation medium 1 by S17, each is made into a pair, and it stores in the software ID storage section 24 and the version ID storage section 25. Then, software 11 is installed in a secondary storage 28 by the installation means 3 (S18).

[0022] In this example (2), by the above-mentioned processing, since the installation ID stored in the software 29 of a secondary storage 28, Software ID, and the same ID as Version ID are registered into the installation ID storage section 23, the software ID storage section 24, and the version ID storage section 25, verification of the consistency of each ID of S42 brings the same result, S44 is performed at the time of execution of a loader 26, and the execution of software 29 of it is

[0023] (3) In the case of the combination which each ID of is storing ending to the installation ID storage section 23, the software ID storage section 24, the version ID storage section 25, and the installation ID storing field 14, and does not have the same ID as the software ID storing field 12 into the software ID storage section 24 That is, it is the case where another software tends to use the software installation medium 1 with which another computer systems were provided for the already installed computer system 2, and tends to copy software 11 to it unjustly.

[0024] In this case, since there is no same software ID as the software ID storing field 12 into the software ID storage section 24, processing is moved to S4 from S3, and new registration processing shown in drawing 3 is performed. Execution of new registration processing advances processing to S15 from S11. Since Installation ID is stored in the installation ID storage section 23 and the installation ID storing field 14, new registration processing is ended. Therefore, software 11 is not installed in a secondary storage 28. However, if the case where software 11 is copied to a secondary storage 28 by a certain method not using the regular software registration method offered by this invention is assumed, when copying first the software memorized by the software installation medium 1 illegally, since software and each ID are one, they are copied to a secondary storage 28 from the software installation medium 1 also about Software ID, Version ID, and Installation ID. For this reason, in the running phase (refer to loader processing of drawing 6) of the copied software, by this example (3), Software ID will not be in agreement with the thing in the software ID storage section 24 at least, and software will terminate abnormally by processing of S42 and S43 of drawing 6 . Consequently, the software copied unjustly cannot be used.

[0025] (4) Although the installation ID storage section 23, the software ID storage section 24, and the version ID storage section 25 have not been ID stored, when ID is stored in the installation ID storing field 14. That is, it is the case where use the software installation medium 1 with which the secondary storage 28 of a computer system 2 was provided to the computer system of the initial state in which software is not installed at another computer system, and software 11 is copied unjustly.

[0026] Also in this case, new registration processing is performed from processing of drawing 2 of S3. Since after new registration processing execution does not have Installation ID in the installation ID storage section 23 in S11, it progresses to S12. Moreover, processing is finished, without starting the installation means 3, since Installation ID is stored in the installation ID storing field 14. Therefore, the software 11 of the software installation medium 1 is not installed in a secondary storage 8. However, though the software 11 of the software installation medium 1 is copied to a secondary storage 28 by a certain method, it will be copied to a secondary storage 28 from the software installation medium 1 also about Software ID, Version ID, and Installation ID. Therefore, since it progresses to S43 from S42 shown in drawing 6 and software terminates abnormally, the software copied unjustly cannot be used.

[0027] (5) In the case of combination with the version ID newer than the version ID in the version ID storage section 25 although each ID is stored in the installation ID storage section 23, the software ID storage section 24, and the version ID storage section 25 which the same ID as the software ID of the software ID storing field 12 which is the combination no ID is [combination] in the installation ID storing field 14 exists in the software ID storage section 24, and is stored in the version ID storing field 13 That is, it is the case where the software 11 in the software installation medium 1 uses the software installation medium 1 offered regularly, and installs software 11 again as upgrade to the already installed computer system 2.

[0028] In this case, since the same ID as the software ID of the software ID storing field 12 exists in the software ID storage section 24, processing progresses to S5 of drawing 2 . The version ID which becomes the software ID which the identification number I/O means 22 acquired Version ID from the version ID storing field 13 to software 11 (S5), and was acquired from the inside of the version ID enclosure within a computer system 2 by S2 next, and a pair is read (S6). Each version ID read by S5 and S6 is compared, and new Koseki charge of the software 11 which will be installed from now on and to carry out, and the software 29 installed in the secondary storage 28 is verified (S7).

[0029] Since the version ID stored in the version ID storing field 13 is newer than the version ID in the version ID storage section 25, it is judged to be upgrade, and this example (5) performs upgrade

processing shown by drawing 4 (S9). If this upgrade processing is performed, the identification number I/O means 22 will process S21-S24 which are shown in drawing 4. In this example (5), since Installation ID has not been stored in the installation ID storing field 14, it judges that Installation ID is not stored in the installation ID storing field 14 by S21, and processing which registers Installation ID to the installation ID storing field 14 by S22 is performed.

[0030] Generating with the installation identification number generation means 21 does not newly carry out installation ID, but the identification number I/O means 22 reads the installation ID already memorized in the installation ID storage section 23 of a computer system 2, and stores this in the installation ID storing field 14 corresponding to the software 11 in the software installation medium 1 here. Next, the version ID in the version ID storage which is the software ID in the software ID storage section 24 which acquired Version ID from the software installation medium 1, and was acquired by S2, and a pair by S23 is updated. Then, the installation means 3 is started by S25, and software 11 is installed in a secondary storage 28.

[0031] In this example (5), the inside of the version ID storage section 25 is updated by the above-mentioned processing using the same version ID as the version ID stored in the software 29 installed in the secondary storage 28. About Installation ID and Software ID, since it has become clear that ID which is already in agreement exists, verification of the consistency of each ID of S42 brings the same result, processing progresses to S44 and the execution of software 29 of it is attained.

[0032] (6) Each ID is stored in the installation ID storage section 23, the software ID storage section 24, the version ID storage section 25, and the installation ID storing field 14. And the same ID as the software ID of the software ID storing field 12 exists in the software ID storage section 24. When the direction of the version ID stored in the version ID storing field 13 is the combination the contents of the installation ID storing field 14 and whose contents of the installation ID storage section 23 is newer than the version ID in the version ID storage section 25, and correspond. That is, once installing the software 11 in the software installation medium 1 in a computer system 2, it is the case where dare install the old version of the same software by a certain reason, use after that the software installation medium 1 currently kept as an object for backup, and software 11 is installed again.

[0033] In this case, it progresses to S5 from S3 shown in drawing 2. In this example (6), although each version ID is acquired and each new Koseki charge is verified by S7 S5 and S6, since the version ID stored in the version ID storing field 13 is newer than the version ID in the version ID storage section 25, it is judged as upgrade and upgrade processing shown by drawing 4 is performed. If this upgrade processing is performed, since ID is stored in the installation ID storing field 14, processing of S24 is performed through S21. In S24, it verifies whether the installation ID in the installation ID storing field 14 and the installation ID in the installation ID storage section 23 are in agreement. In this example (6), since both installations ID are in agreement, after progressing to S23 and updating Version ID, the installation means 3 is started by S25, and software 11 is installed in a secondary storage 28.

[0034] In this example (6), since it has become clear that ID which updates ID of the version ID storage section 25 using the same ID as the version ID stored in the secondary storage 28, and is already in agreement about Installation ID and Software ID with the above-mentioned procedure exists, verification of the consistency of each ID of S42 brings the same result, S44 is performed and the execution of software 29 of it is attained.

[0035] (7) Each ID is stored in the installation ID storage section 23, the software ID storage section 24, the version ID storage section 25, and the installation ID storing field 14. And the same ID as the software ID of the software ID storing field 12 exists in the software ID storage section 24. Although it is newer than the version ID in the version ID storage section 25, when the direction of the version ID stored in the version ID storing field 13 is the combination the content of the installation ID storing field 14 and whose content of the installation ID storage section 23 do not correspond. That is, it is the case where software 11 and software of the same kind tend to use the software installation medium 1 with which another computer systems were provided, and tend to upgrade software 11 unjustly as opposed to the already installed computer system 2. This situation is fully assumed.

[0036] In this case, in S3 shown in drawing 2, since the same software ID as the software ID of the

software ID storing field 12 exists in the software ID storage section 24, both are in agreement and progress to S5. Although each version ID is acquired and each new Koseki charge is verified by S7 S5 and S6, in this example (7), since the version ID stored in the version ID storing field 13 is newer than the version ID in the version ID storage section 25, it is judged to be upgrade, and upgrade processing shown by drawing 4 is performed. If this upgrade processing is performed, since ID is stored in the installation ID storing field 14, it progresses to S24 through S21. In S24 of this example (7), since both installations ID are not in agreement, processing is ended. Therefore, software 11 is not installed in a secondary storage 28. However, though the software 11 of the software installation medium 1 is copied to a secondary storage 28 by a certain method, since the installation ID of the installation ID storing field 14 is copied to a secondary storage 28, Installation ID is not in agreement, an inequality is judged with 26 loaderS42, software cannot terminate abnormally by S43 and the software copied unjustly cannot be used.

[0037] (8) Although each ID is stored in the installation ID storage section 23, the software ID storage section 24, and the version ID storage section 25, when the version ID which the same ID as the software ID of the software ID storing field 12 which is the combination no ID is [combination] in the installation ID storing field 14 exists in the software ID storage section 24, and is stored in the version ID storing field 13, and the version ID in the version ID storage section 25 are equal combination. That is, it is the case where the software 11 in the software installation medium 1 tends to use the software installation medium 1 offered regularly to the already installed computer system 2, and tends to install software 11 doubly accidentally.

[0038] In this case, in this example (8), although it verifies whether each software ID is acquired by S1 and S2, and each is in agreement by S3 like the case of an example (1), since the same software ID as the software ID of the software ID storing field 12 exists in the software ID storage section 24, both are in agreement and processing moves from them to the following S5. Although each version ID is acquired and each new Koseki charge is verified in S5 and S6 S7 as the example (5) showed In this example (8), since the version ID stored in the version ID storing field 13 and the version ID in the version ID storage section 25 are equal, it is judged as re-registration processing of a backup medium, and backup medium registration processing shown by drawing 5 is performed. If this backup medium registration processing is performed, the identification number I/O means 22 will process S31-S33 which are shown in drawing 5 . As shown also in advance, since it judges that Installation ID is stored in S31 since Installation ID is stored in the installation ID storage section 23 and Installation ID has not been stored in the installation ID storing field 14, and it can judge that Installation ID is not stored in S32, processing is ended by this example (8), without starting the installation means 3. Therefore, the software 11 of the software installation medium 1 is not installed in a secondary storage 28.

[0039] Moreover, the software 29 in a secondary storage 28 is destroyed in a certain accident, and this example (8) is considered when saying that offer of the software installation medium 1 was received newly reluctantly in order to restore a system, since the software installation medium which was being kept as an object for backup was also damaged. Then, S32 is ended, and processing is not completed immediately, but the warning message of the purport which is "double registration of the new software installation medium 1" is displayed after S32 end, and you may make it demand from an operator whether installation processing by the installation means 3 of S34 is performed.

[0040] In this example (8), by the above-mentioned processing, since processing of the same software which it was going to install in the secondary storage 28 being already installation ending as software 29, and updating each ID is not performed, verification of each ID of S42 becomes the same, S44 is performed at the time of execution of a loader 26, and the execution of software 29 of it is attained.

[0041] (9) Each ID is stored in the installation ID storage section 23, the software ID storage section 24, the version ID storage section 25, and the installation ID storing field 14. And the version ID by which the same ID as the software ID of the software ID storing field 12 exists in the software ID storage section 24, and is stored in the version ID storing field 13 When it is the combination the contents of the installation ID storing field 14 and whose contents of the installation ID storage section 23 is equal, and correspond. [of the version ID in the version ID storage section 25] That is, although installation of

software 11 was performed by the software installation medium 1 with which the computer system 2 was provided regularly, since the software 29 of a secondary storage 28 carried out destruction etc., it is the case where use the same software installation medium 1 which has become backup, and re-installation is performed.

[0042] In this case, although it verifies to see does it acquire each software ID in S1 and S2 which are shown in drawing 2 and each is in agreement by S3, in this example (9), since the same software ID as the software ID of the software ID storing field 12 exists in the software ID storage section 24, both are in agreement, and it moves from processing to S5. And in this example (9), although each version ID is acquired by S5 and S6 and each new Koseki charge is verified by S7, since the version ID stored in the version ID storing field 13 and the version ID in the version ID storage section 25 are equal, it is judged as re-registration processing of a backup medium, and backup medium registration processing shown by drawing 5 is performed. If this backup medium registration processing is performed, since Installation ID is stored in the installation ID storage section 23 and the installation ID storing field 14, both installations ID are compared by S33 through S31 and S32. In S33, it verifies whether the installation ID in the installation ID storing field 14 and the installation ID in the installation ID storage section 23 are in agreement. Since both installations ID are in agreement, it progresses to S34, and the installation means 3 is started, and software 11 is made to install in a secondary storage 28 in this example (9).

[0043] The installation ID of as opposed to [at this example (9)] the software 11 in the software installation medium 1 by the above-mentioned processing Since it has become clear that all the software ID and versions ID are in agreement with ID stored in each of the installation ID storage section 23, the software ID storage section 24, and the version ID storage 15 Verification of the consistency of each ID of S42 brings the same result, S44 is performed at the time of execution of a loader 26, and the execution of software 29 of it is attained.

(10) Each ID is stored in the installation ID storage section 23, the software ID storage section 24, the version ID storage section 25, and the installation ID storing field 14. And the version ID by which the same ID as the software ID of the software ID storing field 12 exists in the software ID storage section 24, and is stored in the version ID storing field 13 When it is the combination the contents of the installation ID storing field 14 and whose contents of the installation ID storage section 23 is equal, and do not correspond. [of the version ID in the version ID storage section 25] That is, it is the case where the same software as the software 11 in the software installation medium 1 tends to use the software installation medium 1 with which another computer systems were provided as opposed to the computer system 2 already installed in the secondary storage 28 of a computer system 2, and tends to restore software 11 from a backup medium unjustly.

[0044] In this case, although it verifies whether each software ID is acquired by S1 and S2, and each is in agreement by S3, in this example (10), since the same software ID as the software ID of the software ID storing field 12 exists in the software ID storage section 24, both are in agreement, and it moves from processing to S5. Although each version ID is acquired like the case where an example (5) shows and each new Koseki charge is verified in S5 and S6 S7 In this example (10), since the version ID stored in the version ID storing field 13 and the version ID in the version ID storage section 25 are equal, it is judged as re-registration processing of a backup medium, and backup medium registration processing shown by drawing 5 is performed. If this backup medium registration processing is performed, since it can judge that both installations ID are not in agreement by S33 through S31 and S32, processing is ended without starting the installation means 3. Therefore, the software 11 of the software installation medium 1 is not installed in a secondary storage 28.

[0045] Moreover, though the software 11 of the software installation medium 1 is copied to a secondary storage 28, since the installation ID of the installation ID storing field 14 is copied to a secondary storage 28 from the software installation medium 1 by a certain method, Installation ID is not in agreement, it is judged by processing of 26 loaderS42, software can terminate abnormally by processing of S43, and the software copied unjustly cannot be used.

[0046] (11) In the case of combination with the version ID older than the version ID in the version ID storage section 25 although each ID is stored in the installation ID storage section 23, the software ID

storage section 24, and the version ID storage section 25 by which is been ID non-stored combination in the installation ID storing field 14, and the same ID as the software ID of the software ID storing field 12 exists in the software ID storage section 24, and is stored in the version ID storing field 13 That is, although it is the software installation medium 1 offered regularly as an object for new installation, it is the case where it is going to install the software of a version older than the software [finishing / installation] 29 to the computer system 2 in which an installation place computer system is mistaken and the software of the same kind is already installed.

[0047] In this case, although it verifies to see does it acquire each software ID by S1 and S2 and each is in agreement by S3, in this example (11), since the same software ID as the software ID of the software ID storing field 12 exists in the software ID storage section 24, both are in agreement, and it moves from processing to S5. In S5 and S6, although each version ID is acquired and each new Koseki charge is verified by S7, since the version ID stored in the version ID storing field 13 is older than the version ID in the version ID storage section 25, in this example (11), it is judged that there is no need for installation, and it ends processing, without starting the installation means 3.

[0048] Therefore, the software 11 of the software installation medium 1 is not installed in a secondary storage 28. moreover -- in such a case, although the version of software will become old, since it is the installation processing which uses the software installation medium 1 offered regularly, the situation that a certain need of daring return the software 29 in a secondary storage 28 to an old version for convenience' sake arose is also considered Then, S7 is ended, and processing is not completed immediately, but the warning message of the purport "makes the version of the software in a secondary storage old" after S7 end is displayed, and you may make it demand from an operator whether installation processing by the installation means 3 is performed. The installation processing in this case becomes equivalent to the upgrade processing shown by drawing 4 . If upgrade processing temporarily shown by drawing 4 is carried out, installation processing will be carried out by the installation means 3 of S25 through S21, S22, and S23.

[0049] (12) In the case of combination with the version ID older than the version ID in the version ID storage section 25 by which each ID is stored in the installation ID storage section 23, the software ID storage section 24, the version ID storage section 25, and the installation ID storing field 14, and the same ID as the software ID of the software ID storing field 12 exists in the software ID storage section 24, and is stored in the version ID storing field 13 That is, in case the software installation medium 1 which has become backup is used and re-installation of software 11 is performed since the software 29 of a secondary storage 28 carried out destruction etc. when upgrade processing is performed to the software 29 in the secondary storage 28 of a computer system 2 and the software installation medium 1 is kept as a medium for backup for computer system 2, it is the case where it is said that the backup medium of an old version has been used accidentally.

[0050] In this case, although it verifies to see does it acquire each software ID by S1 and S2 and each is in agreement by S3, in this example (12), since the same ID as the software ID of the software ID storing field 12 exists in the software ID storage section 24, both are in agreement, and it moves from processing to S5. In S5 and S6, although each version ID is acquired and each new Koseki charge is verified by S7, since the version ID stored in the version ID storing field 13 is older than the version ID in the version ID storage section 25, it judges that there is no need for installation, and processing is ended, without starting the installation means 3.

[0051] Therefore, the software 11 of the software installation medium 1 is not installed in a secondary storage 28. moreover -- in such a case, since it is the installation processing which uses the software installation medium 1 for backup, the situation that a certain need of daring return the software 29 in a secondary storage 28 to an old version for convenience' sake arose is also considered Then, S7 is ended, and processing is not completed immediately, but the warning message of the purport "makes the version of the software in a secondary storage old" after S7 end is displayed, and you may make it urge whether installation processing by installation **** 7 is performed to the operation rhe evening. The installation processing in this case becomes equivalent to the upgrade processing shown by drawing 4 . If upgrade processing temporarily shown by drawing 4 is carried out, installation processing will be

carried out by the installation means 3 of S25 through S21, S22, and S23.

[0052] (13) Each ID is stored in the installation ID storage section 23, the software ID storage section 24, the version ID storage section 25, and the installation ID storing field 14. And the same ID as the software ID of the software ID storing field 12 exists in the software ID storage section 24. Although it is older than the version ID in the version ID storage section 25, when the direction of the version ID stored in the version ID storing field 13 is the combination the contents of the installation ID storing field 14 and whose contents of the installation ID storage section 23 do not correspond. That is, since the software 29 of the secondary storage 28 of a computer system 2 carried out destruction etc., in case the software installation medium 1 which has become backup is used and re-installation of software 11 is performed, it is the case where the backup medium of an old version currently kept as another object for computer systems has been used accidentally.

[0053] In this case, in S1 and S2 which are shown in drawing 2, although it verifies whether each software ID is acquired and each is in agreement by S3, in this example (13), since the same software ID as the software ID of the software ID storing field 12 exists in the software ID storage section 24, both are in agreement, and it moves from processing to S5. Although each version ID is acquired and each new Koseki charge is verified by S7 S5 and S6, since the version ID stored in the version ID storing field 13 is older than the version ID in the version ID storage section 25, it is judged that there is no need for installation, and it ends processing, without starting the installation means 3.

[0054] Therefore, the software 11 of the software installation medium 1 is not installed in a secondary storage 28. Moreover, processing is ended, without starting the installation means 3, since Installation ID is not in agreement by S24 through S21 even if it carries out upgrade processing shown by drawing 4 like an example (11) and (12). Therefore, though the software 11 of the software installation medium 1 can be copied to a secondary storage 28 Since the installation ID of the installation ID storing field 14 is copied to a secondary storage 28 from the software installation medium 1, in this example (13) Installation ID is not in agreement, it is judged by processing of 26 loaderS42, software can terminate abnormally by processing of S43, and the software copied unjustly cannot be used.

[0055] As mentioned above, although 1 operation form of this invention was explained, this invention is not limited to this, in addition various kinds of addition change is possible for it. For example, although the number of the software stored in the software installation medium 1 is one in this embodiment, since the version ID which shows the software ID showing the classification of software and a version is formed, it is also possible to store two or more software in a mass medium.

[0056] Moreover, when using the medium which cannot do re-writing like a CD-ROM medium, the software without the need of writing nothing in itself can be stored in a CD-ROM medium, and it can respond by forming an installation common medium using the medium which can be written in, for example, FD medium, and preparing an installation ID storing field there about the installation ID which write-in processing generates.

[0057] Drawing 7 showed the form of this operation. The identification number I/O means 220 starts CD-ROM access **** 410 in the installation common medium 400, reads Software ID and Version ID from the software ID storing field 111 in the CD-ROM medium 100, and the version ID storing field 112, and the installation method in this case displays the registration software list in the CD-ROM medium 100, and installs which software, or urges it to directions at an operator.

[0058] If it is determined what software is installed, in order to discriminate the selected software, selected Software ID and selected Version ID of software are held in the CD-ROM access means 410. The installation identification number generation means 210 receives Installation ID from the installation ID storage section 221, and the identification number I/O means 220 stores it in the installation ID storing field 411 in the installation common medium 400, after checking that the installation ID for other computer systems is not stored in the installation ID storing field 411.

[0059] Then, the software ID stored in the software ID storing field 111 of software 110 and the version ID storing field 112 and Version ID are stored in the software ID storage section 222 and the version ID storage section 223 which are formed in BUM within a computer system 200, respectively, and the installation means 300 is started. The installation means 300 starts the CD-ROM access means 410, in

order to install applicable software into a secondary storage 280 from the CD-ROM medium 100, if this demand is received. The CD-ROM access means 410 will copy the software 110 in the CD-ROM medium 100 to a secondary storage 280 based on Software ID and Version ID which were held previously, if starting from the installation means 300 is received.

[0060] If the copy of software 110 is completed, the installation means 300 will copy the content of the installation ID storing field 411 in the installation common medium 400 to the installation ID storage region 2830 in a secondary storage 280. Consequently, even if it cannot write the direct installation ID in the CD-ROM medium 100 used as a software installation medium, the software 110 grade which recorded each ID on the secondary storage 280 is installable. Moreover, since it can judge that the identification number I/O means 220 verifies the inside of the installation ID storing field 411 at the time of initial processing of installation, and it is not in agreement with the installation ID of the installation ID storage section 221 at it when the installation common medium with which other computer systems are provided is used, an unjust copy can be prevented. Moreover, though software can be copied to injustice from the CD-ROM medium 100 at malice, since the installation ID storage region 2830 is in a blank state, software cannot be performed.

[0061]

[Effect of the Invention] Software can be prevented from an unjust copy, securing backup according to this invention, since other computer cis-TEMUHE is not installable using the software installation medium even if it leaves the software installation medium offered for a certain computer system, so that clearly from the above explanation. Moreover, the same software is accidentally installed in a computer system doubly, and it is effective in the ability to prevent that it becomes impossible to use it by the alien system.

[0062] Furthermore, if two or more software is registered using the mass medium, it is not necessary to generate a medium for every classification of software, it becomes possible to carry out package mass production method, and reduction of a production cost can be realized.

[Translation done.]

* NOTICES *

Japan Patent Office is not responsible for any damages caused by the use of this translation.

1. This document has been translated by computer. So the translation may not reflect the original precisely.
2. **** shows the word which can not be translated.
3. In the drawings, any words are not translated.

DESCRIPTION OF DRAWINGS

[Brief Description of the Drawings]

[Drawing 1] this invention is a block diagram [like] 1 operative condition.

[Drawing 2] Procedure explanatory drawing of the common processing at the time of software installation.

[Drawing 3] Procedure explanatory drawing at the time of new registration of software.

[Drawing 4] Procedure explanatory drawing in the case of being software at the upgrade time.

[Drawing 5] Procedure explanatory drawing in the case of installing software from a backup medium.

[Drawing 6] Procedure explanatory drawing of a software running phase.

[Drawing 7] The block diagram when registering two or more software into the medium of one sheet.

[Description of Notations]

1 Software Installation Medium

2,200 Computer system

11 29,290 Software

3,300 Installation means

12,111 Software ID storing field

13,112 Version ID storing field

14,113,411 Installation ID storing field

21,210 Installation identification number generation means

22,220 Installation identification number I/O means

23,221 Installation ID storage section

24,222 Software ID storage section

25,223 Version ID storage section

26,226 Loader

27,227 Identification number comparison means

28,280 Secondary storage

100 CD-ROM Medium

281 2810 Software ID storage region

282 2820 Version ID storage region

283 2830 Installation ID storage region

400 Installation Common Medium

410 CD-ROM Access Means

[Translation done.]

* NOTICES *

Japan Patent Office is not responsible for any damages caused by the use of this translation.

1. This document has been translated by computer. So the translation may not reflect the original precisely.
2. **** shows the word which can not be translated.
3. In the drawings, any words are not translated.

CLAIMS

[Claim(s)]

[Claim 1] The method installed in the storage of the computer system which has non-volatile memory for the software stored in the storage characterized by providing the following. The installation ID storing field which stores the installation ID which expresses the existence of installation execution of the aforementioned software with the aforementioned storage, While forming the software ID storing field which stores the software ID which shows the classification of software, and the version ID storing field which stores the version ID which shows the version of the software concerned An installation ID generating means to form the storage region of the aforementioned installation ID, the aforementioned software ID, and the aforementioned version ID in the aforementioned non-volatile memory, and to generate the aforementioned installation ID in the aforementioned computer system further at the time of installation of the beginning of the aforementioned software. The means which reads the software ID which corresponds from the aforementioned software ID storing field, writes in the software ID storage region of the aforementioned non-volatile memory, reads further the version ID which corresponds from the aforementioned version ID storing field, and is written in the version ID storage region of the aforementioned non-volatile memory while writing the generated installation ID in the installation ID storing field of the aforementioned storage, and the installation ID storage region of the aforementioned non-volatile memory.

[Claim 2] The installation method according to claim 1 which carries out [having established a means verify consistency with each ID memorized by each ID stored in the aforementioned storage, and each aforementioned storage region, and determine the execution propriety of the aforementioned installation based on this verification result, when the aforementioned software is installed in it, while forming the storage region of the aforementioned installation ID, the aforementioned software ID, and the aforementioned version ID in the storage of the aforementioned computer system, and] as the feature.

[Claim 3] The aforementioned computer system is an installation method according to claim 1 or 2 characterized by having an interface means to display a predetermined message in the case of installation of the aforementioned software, and to receive the response result of this message further, and determining the execution propriety of installation based on this received response result.

[Claim 4] The method installed in the storage of the computer system which has non-volatile memory for the software stored in the storage characterized by providing the following: Equip the aforementioned storage with the common medium in which accessible rewriting is free, and the installation ID storing field which stores the installation ID which expresses the existence of installation execution of the aforementioned software with this common medium at least is formed. While forming the version ID storing field which stores the version ID which shows the version of the software ID storing field which stores the software ID which shows the classification of the aforementioned software to the aforementioned storage, and the software concerned An installation ID generating means to form the storage region of the aforementioned installation ID, the aforementioned software ID, and the aforementioned version ID in the aforementioned non-volatile memory, and to generate the aforementioned installation ID in the aforementioned computer system further at the time of installation

of the beginning of the aforementioned software. The means which reads the software ID which corresponds from the aforementioned software ID storing field, writes in the software ID storage region of the aforementioned non-volatile memory, reads further the version ID which corresponds from the aforementioned version ID storing field, and is written in to the version ID storage region of the aforementioned non-volatile memory while writing the generated installation ID in the installation ID storing field of the aforementioned common medium, and the installation ID storage region of the aforementioned non-volatile memory.

[Claim 5] It is the real whereabouts formula of the software installed in the storage of the computer system which has non-volatile memory. The installation ID storage region which memorizes the installation ID which expresses the existence of installation execution of the aforementioned software to the aforementioned storage, While forming the software ID storage region which memorizes the software ID which shows the classification of software, and the version ID storage region which memorizes the version ID which shows the version of the software concerned The aforementioned installation ID inputted into the aforementioned non-volatile memory at the time of installation of the aforementioned software The storage region of the aforementioned software ID and the aforementioned version ID is formed. Furthermore, the real whereabouts formula of the software characterized by establishing a means to verify the consistency of each ID stored in each aforementioned storage region, and each ID stored in the aforementioned non-volatile memory to the aforementioned computer system, and to determine the execution propriety of the aforementioned software as it based on this verification result.

[Translation done.]